Amendments to the Claims:

The following listing of claims will replace all prior versions, and listings, of claims in the application:

1-4. (Canceled)

5. (Currently Amended) A method for performing parallel sorting processing using a parallel processor, where (i) one unsorted data string, or (ii) two or more sorted partial data strings are received as an input, the method comprising:

if the input is one unsorted data string, a first step of dividing the unsorted data string to generate a plurality of unsorted partial data strings;

a second step of assigning a processor to said plurality of unsorted partial data strings respectively;

a third step of sorting each of said plurality of unsorted partial data strings independently by the assigned processor based on an arbitrary algorithm, to generate sorted partial data strings, the arbitrary algorithm sorting the plurality of unsorted partial data strings ~~being sorted~~ each in one of either ascending or descending order;

a fourth step of creating an input data string pair for sorting using two of the sorted partial data strings which were acquired in the third step or a seventh step or that were received as the input, and dividing the pair into sets of partial data string pairs respectively under a predetermined division condition;

a fifth step of editing control information relating to parallel processing of merge processing, the control information for parallel sorting of the plurality of divided partial data string pairs, the control information being information regarding at least the location of the plurality of divided partial data string pairs and the number of the plurality of divided partial data string pairs;

a sixth step of assigning to the plurality of divided partial data string pairs a processor group for merging of the plurality of divided partial data string pairs;

the seventh step of performing merge processing by the assigned processor and outputting sorted partial data strings; and

a step of repeating said fourth step to said seventh step using the merge-processed sorted partial data strings as said sorted partial data strings, wherein the above steps are applied when one unsorted data string is provided, and the first to the third steps are unnecessary if two or more sorted data strings are provided, and a repeat of said fourth to seventh steps ends when the merge-processed sorted partial data strings are merged into one data string;

if only two sorted data strings are provided, the fourth to seventh steps are executed only once, and the repeat is not~~ required.~~ required,

wherein the following items are satisfied as the predetermined division condition in said fourth step,

when a condition where data string pair (D_1, n) and (D_2, n) are divided into two partial data string pairs $\{(D_{11}, n_{11}), (D_{21}, n_{21})\}$ and $\{(D_{12}, n_{12}), (D_{22}, n_{22})\}$, is established, and also

$$n_{11} + n_{21} = 2x, n_{12} + n_{22} = 2(n - x)$$

is established, where x is a half value of the number of data of the partial data string pair $\{(D_{11}, n_{11}), (D_{21}, n_{21})\}$, and is also the number of data of D_{11} and D_{21} when $n_{11} = n_{21}$, and

wherein said fourth step has the following functions:

(1) an operation to divide a sorted data string pair $\{(D_1, n), (D_2, n)\}$ into k sets of segment pairs, which is equivalent to performing (k-1) sets of two-division operations in which a total of the number of data counted from a first part of D_1 and D_2 becomes 2x with changing a value of x, while considering a magnitude of the key values of both data strings; in

this case, a sub-division problem of the sorted data string pair $\{(D_1, n), (D_2, n)\}$ to the k sets of segment pairs is replaced with the above-mentioned two-division problem of the data string that satisfies the items as the division condition in the fourth step;

(2) specifying a data position in the data string by an index value, the value sequentially increments with the index value of a first data in the data string D_1 or D_2 as 0, x indicates the number of data, but if the value of x itself is regarded as an index value, then [x] indicates the (x+1)th data counted as 1, 2, 3, . . . from the first part of the data string;

if $n_{11} = n_{21}$, then $n_{11} = n_{21} = x$, which is a formula indicating the number of data, can be interpreted that the position of the xth data counted from the first part, that is data with the index value x-1, is at a division boundary of D_1 and D_2;

(3) an area dividing function, comprising:

a step of setting said x as an initial value of a boundary index value for index variables i and j for specifying individual data in said data strings D_1 and D_2;

a comparison step of comparing a key value of data indicated by the index variable i of the data string D_1 and a key value of data indicated by the index variable j of the data string D_2;

a step of adding 1 to an index variable of the data with a greater key value, subtracting 1 from an index variable of the data with a smaller key value, then branching processing to said comparison step, if the key value of the data indicated by the index variable i of D_1 and the key value of the data indicated by the index variable j of D_2 are not the same in an initial comparison;

a step of adding 1 to the index variable of data with a greater key value, and subtracting 1 from the index variable of data with a smaller key value, then branching processing to said comparison step, if a magnitude relationship of the key value of data

indicated by the index variable i of D_1 and the key value of data indicated by the index variable j of D_2 is unchanged in a second or later comparison;

a step of regarding the data indicated by the index variable i and the data indicated by the index variable j as a division boundary respectively, if the key value of the data indicated by the index variable i of D_1 and the key value of the data indicated by the index variable j of D_2 are the same in the initial comparison; and

a step of comparing a greater one of the key value of D_1 and the key value of D_2 in a previous comparison operation with the greater one of the key value of D_1 and the key value of D_2 in a current comparison operation, and regarding the data with a smaller key value as the division boundary and regarding the data initially compared with this data as the other boundary, if the magnitude relationship between the key value of the data indicated by the index variable i of D_1 and the key value of the data indicated by the index variable j of D_2 is inverted from a previous magnitude relationship.

6. (Previously Presented) The parallel sorting processing method according to Claim 5, wherein two processors are assigned to said partial data string pair in said sixth step, and the control information relating to parallel processing of merge processing is edited in the fifth step so that a first processor performs merge processing in descending order from a side of which a key value is greater in said partial data string and a second processor performs merge processing in ascending order from an edge of which a key value is smaller in the same partial data string simultaneously in said seventh step.

7-10. (Canceled)

11. (Currently Amended) A computer readable storage medium that stores a program for performing parallel sorting processing using a parallel processor which includes a plurality of processors, where (i) one unsorted data string, or (ii) two or more sorted partial data strings are received as an input, the program comprising instructions that execute:

if the input is one unsorted data string, a first step of dividing the unsorted data string to generate a plurality of unsorted partial data strings;

a second step of assigning a processor to said plurality of unsorted partial data strings respectively;

a third step of sorting each of said plurality of unsorted partial data strings independently by the assigned processor based on an arbitrary algorithm, to generate sorted partial data strings, <u>the arbitrary algorithm sorting </u>the plurality of unsorted partial data strings ~~being sorted~~ each in one of either ascending or descending order;

a fourth step of creating an input data string pair for sorting using two sorted partial data strings which were generated in the third step or a seventh step or that were received as the input, and dividing the pair into sub-divided partial data string pairs respectively under a predetermined division condition;

a fifth step of editing control information relating to parallel processing of merge processing, the control information for sorting of the divided partial data string pairs, <u>the control information being information regarding at least the location of the plurality of divided partial data string pairs and the number of the plurality of divided partial data string pairs</u>;

a sixth step of assigning a processor group to each of the sub-divided partial data string pairs;

the seventh step of performing merge processing in parallel by the assigned processors and outputting sorted partial data strings; and

a step of repeating said fourth step to said seventh step using the merge-processed data strings as said partial data strings, wherein the above steps are applied when one unsorted data string is provided, and the first step to the third step are unnecessary if two

or more sorted data strings are provided, and a repeat of said fourth step to said seventh step ends when the merge-processed sorted partial data strings become one data string;

the fourth step to the seventh step are executed once, and the repeat is not required if two sorted data strings are provided.

wherein the following items are satisfied as the predetermined division condition in said fourth step,

when a condition where data string pair (D_1, n) and (D_2, n) are divided into two partial data string pairs $\{(D_{11}, n_{11}), (D_{21}, n_{21})\}$ and $\{(D_{12}, n_{12}), (D_{22}, n_{22})\}$, is established, and also

$$n_{11} + n_{21} = 2x, n_{12} + n_{22} = 2 (n - x)$$

is established, where x is a half value of the number of data of the partial data string pair $\{(D_{11}, n_{11}), (D_{21}, n_{21})\}$, and is also the number of data of D_{11} and D_{21} when $n_{11} = n_{21}$, and

wherein said fourth step has the following functions:

(1) an operation to divide a sorted data string pair $\{(D_1, n), (D_2, n)\}$ into k sets of segment pairs, which is equivalent to performing (k-1) sets of two-division operations in which a total of the number of data counted from a first part of D_1 and D_2 becomes 2x with changing a value of x, while considering a magnitude of the key values of both data strings; in this case, a sub-division problem of the sorted data string pair $\{(D_1, n), (D_2, n)\}$ to the k sets of segment pairs is replaced with the above-mentioned two-division problem of the data string that satisfies the items as the division condition in the fourth step;

(2) specifying a data position in the data string by an index value, the value sequentially increments with the index value of a first data in the data string D_1 or D_2 as 0, x indicates the number of data, but if the value of x itself is regarded as an index value, then [x] indicates the (x+1)th data counted as 1, 2, 3, . . . from the first part of the data string;

if $n_{11} = n_{21}$, then $n_{11} = n_{21} = x$, which is a formula indicating the number of data, can be interpreted that the position of the xth data counted from the first part, that is data with the index value x-1, is at a division boundary of D_1 and D_2;

(3) an area dividing function, comprising:

a step of setting said x as an initial value of a boundary index value for index variables i and j for specifying individual data in said data strings D_1 and D_2;

a comparison step of comparing a key value of data indicated by the index variable i of the data string D_1 and a key value of data indicated by the index variable j of the data string D_2;

a step of adding 1 to an index variable of the data with a greater key value, subtracting 1 from an index variable of the data with a smaller key value, then branching processing to said comparison step, if the key value of the data indicated by the index variable i of D_1 and the key value of the data indicated by the index variable j of D_2 are not the same in an initial comparison;

a step of adding 1 to the index variable of data with a greater key value, and subtracting 1 from the index variable of data with a smaller key value, then branching processing to said comparison step, if a magnitude relationship of the key value of data indicated by the index variable i of D_1 and the key value of data indicated by the index variable j of D_2 is unchanged in a second or later comparison;

a step of regarding the data indicated by the index variable i and the data indicated by the index variable j as a division boundary respectively, if the key value of the data indicated by the index variable i of D_1 and the key value of the data indicated by the index variable j of D_2 are the same in the initial comparison; and

a step of comparing a greater one of the key value of D_1 and the key value of D_2 in a previous comparison operation with the greater one of the key value of D_1 and the key

value of D_2 in a current comparison operation, and regarding the data with a smaller key value as the division boundary and regarding the data initially compared with this data as the other boundary, if the magnitude relationship between the key value of the data indicated by the index variable i of D_1 and the key value of the data indicated by the index variable j of D_2 is inverted from a previous magnitude relationship.

12. (New) A method for performing parallel sorting processing using a parallel processor, where (i) one unsorted data string, or (ii) two or more sorted partial data strings are received as an input, the method comprising:

if the input is one unsorted data string, a first step of dividing the unsorted data string to generate a plurality of unsorted partial data strings;

a second step of assigning a processor to said plurality of unsorted partial data strings respectively;

a third step of sorting each of said plurality of unsorted partial data strings independently by the assigned processor based on an arbitrary algorithm, to generate sorted partial data strings, the arbitrary algorithm sorting the plurality of unsorted partial data strings each in one of either ascending or descending order;

a fourth step of creating an input data string pair for sorting using two of the sorted partial data strings which were acquired in the third step or a seventh step or that were received as the input, and dividing the pair into sets of partial data string pairs respectively under a predetermined division condition;

a fifth step of editing control information relating to parallel processing of merge processing, the control information for parallel sorting of the plurality of divided partial data string pairs, the control information being information regarding at least the location of the plurality of divided partial data string pairs and the number of the plurality of divided partial data string pairs;

a sixth step of assigning to the plurality of divided partial data string pairs a processor group for merging of the plurality of divided partial data string pairs;

the seventh step of performing merge processing by the assigned processor and outputting sorted partial data strings; and

a step of repeating said fourth step to said seventh step using the merge-processed sorted partial data strings as said sorted partial data strings, wherein the above steps are applied when one unsorted data string is provided, and the first to the third steps are unnecessary if two or more sorted data strings are provided, and a repeat of said fourth to seventh steps ends when the merge-processed sorted partial data strings are merged into one data string;

if only two sorted data strings are provided, the fourth to seventh steps are executed only once, and the repeat is not required,

wherein the following items are satisfied as the predetermined division condition in said fourth step,

when a condition where data string pair (D_1, n) and (D_2, n) are divided into two partial data string pairs $\{(D_{11}, n_{11}), (D_{21}, n_{21})\}$ and $\{(D_{12}, n_{12}), (D_{22}, n_{22})\}$, is established, and also

$$n_{11} + n_{21} = 2x,\ n_{12} + n_{22} = 2\ (n - x)$$

is established, where x is a half value of the number of data of the partial data string pair $\{(D_{11}, n_{11}), (D_{21}, n_{21})\}$, and is also the number of data of D_{11} and D_{21} when $n_{11} = n_{21}$, and

wherein said fourth step has the following functions:

(1) an operation to divide a sorted data string pair $\{(D_1, n), (D_2, n)\}$ into k sets of segment pairs, which is equivalent to performing (k-1) sets of two-division operations in which the total of the number of data counted from the first part of D_1 and D_2 becomes 2x

with changing the value of x, while considering a magnitude of key values of both data strings;

(2) specifying a data position of the data string by an index value, the value sequentially increments with the index value of a first data in D_1 or D_2 as 0, x indicates the number of data, but if the value of x itself is regarded as an index value, then [x] indicates the (x+1)th data counted from the first part in the data strings;

if $n_{11} = n_{21}$, then $n_{11} = n_{21} = x$ indicates that the division boundary of D_1 and D_2 exists at the position of the xth data counted from a first part, that is at the data position of the index value x-1;

(3) an area division function, comprising:

a step of setting said x-1 as an initial value of a boundary index value for index variables i and j for specifying individual data in said data strings D_1 and D_2;

a comparison step of comparing a key value of data indicated by the index variable i of the data string D_1 and a key value of data indicated by the index variable j of the data string D_2;

a step of adding 1 to an index variable of the data with a greater key value, subtracting 1 from an index variable of the data with a smaller key value, then branching processing to said comparison step, when the key value of the data indicated by the index variable i of D_1 and the key value of the data indicated by the index variable j of D_2 are not the same in an initial comparison;

a step of adding 1 to an index variable of the data with a greater key value, subtracting 1 from an index variable of the data with a smaller key value, then branching processing to said comparison step, if a magnitude relationship of the key value of the data indicated by the index variable i of D_1 and the key value of the data indicated by the index variable j of D_2 is unchanged in the second or later comparison;

a step of regarding the data indicated by the index variable i and the data indicated by the index variable j as a division boundary respectively, when the key value of the data indicated by the index variable i of D_1 and the key value of the data indicated by the index variable j of D_2 are the same in the initial comparison; and

a step of comparing the smaller one of the key value of D_1 and the key value of D_2 in the previous comparison operation with the smaller one of the key value of D_1 and the key value of D_2 in the current comparison operation, and regarding the data with a greater key value as a division boundary, and regarding the data initially compared with this data as another boundary, if the magnitude relationship between the key value of the data indicated by the index variable i of D_1 and the key value of the data indicated by the index variable j of D_2 is inverted from a previous magnitude relationship.